ABSTRACT

The area of content-based image retrieval is a hybrid research area that requires knowledge of both computer vision and of database systems. Users are exploiting the opportunity to access remotely-stored images in all kinds of new and exciting ways. However, this has exacerbated the problem of locating a desired image in a large and varied collection. This has led to the rise of a new research and development field known as content-based image retrieval (CBIR), the retrieval of images on the basis of features automatically extracted from the images themselves.

Content-based retrieval systems utilize measures that are based on low-level attributes of the image itself, including color histograms, color composition, and texture. State-of-the-art research focuses on more powerful measures that can find regions of an image corresponding to known objects that users wish to retrieve.

Our research was focus on a unified methodology for feature representation and object class recognition. The aim of this project was to develop an image retrieval method that utilizes the layout and the structure of the perceptually correct color within an image to measure the similarity of images. The developed methodology can be used as automatic indexing capabilities for large image databases.
TABLE OF CONTENTS

Abstract ........................................................................................................................................... ii
Table of Contents ............................................................................................................................ iii
List of Figures ................................................................................................................................... v
List of Tables ..................................................................................................................................... vi
1. Background and Rationale ........................................................................................................... 1
   1.1 Introduction .............................................................................................................................. 1
   1.2 Growth of Digital Imaging ....................................................................................................... 1
   1.3 Need for Image Data Management .......................................................................................... 2
   1.4 Characteristics of Image Queries ............................................................................................ 3
   1.5 Current Content-Based Image Retrieval Techniques .............................................................. 4
       1.5.1 Color Retrieval .................................................................................................................. 5
       1.5.2 Texture Retrieval ............................................................................................................. 5
       1.5.3 Shape Retrieval ................................................................................................................. 6
       1.5.4 Retrieval by Other Types of Primitive Feature ................................................................. 7
   1.6 Objective of the Project ............................................................................................................ 8
   1.7 Application Area of Content Based Image Retrieval .............................................................. 9
       1.7.1 Crime Prevention ............................................................................................................... 9
       1.7.2 The Military ..................................................................................................................... 10
       1.7.3 Intellectual Property ........................................................................................................ 10
       1.7.4 Architectural and Engineering Design ............................................................................. 10
       1.7.5 Fashion and Interior Design ............................................................................................. 10
1.7.6 Journalism and Advertising........................................... 11
1.7.7 Medical Diagnosis.................................................. 11
1.7.8 Geographical Information Systems and Remote Sensing...... 11
1.7.9 Cultural Heritage.................................................. 12
1.7.10 Education and Training......................................... 12
1.7.11 Home Entertainment............................................ 12
1.7.12 Web Searching.................................................... 12

2. Narrative.............................................................................. 13

2.1 Current Research Trends............................................ 13
2.2 Research and Developed System................................. 15

3. System Design and Implementation.................................. 18

3.1 Fundamental of Images............................................. 18

3.1.1 Definition............................................................. 18
3.1.2 RGB Color Model................................................. 19
3.1.3 Histograms.......................................................... 20

3.2 CBIR Using a Rough Set with Heuristics....................... 20

3.3 Developed System Design........................................... 21

3.3.1 Histogram Generation & Pixel Grabber.................... 21
3.3.2 Heuristics Techniques & Rough Set Comparison........... 23
3.3.3 Membership Function for Image Grading.................... 25

4 Results and Discussions.................................................. 27

4.1 Results Obtained....................................................... 27
Testing and Evaluation.................................................................34

5.1 System Performance Metrics..................................................34

5.2 Unit Testing...............................................................................34

5.3 Regression Testing.................................................................34

Conclusion......................................................................................36

6.1 Conclusion...............................................................................36

6.2 Limitations................................................................................36

Bibliography and References..........................................................37

APPENDIX A...............................................................................40

APPENDIX B...............................................................................50
LIST OF FIGURES

Figure 1.1 Diagram for Content Based Image Retrieval System...............................4
Figure 2.1 Content Based Image Retrieval Framework........................................14
Figure 2.2 A Generic CBIR Model........................................................................16
Figure 3.1 Digitization of a Continuous Image......................................................18
Figure 3.2 RGB Color Model..............................................................................19
Figure 3.3 Roses and its Histogram......................................................................20
Figure 3.4 Level 1 Data Flow Diagram..................................................................22
Figure 3.5 Level 1 Data Flow Diagram..................................................................24
LIST OF TABLES

Table 3.1 Decision Table for Training Set......................................................23
Table 4.1 Category: Neon Images.................................................................27
Table 4.2 Category: Rose Images.................................................................27
Table 4.3 Category: Animal Images.............................................................28
Table 4.4 Set of Uncategorized Images.........................................................28
Table 4.5 Membership Values for Uncategorized Images...............................32
Table 4.6 Rating Given to New Images by System.........................................33
1. BACKGROUND AND RATIONALE

1.1 Introduction

Content-based image retrieval, a technique which uses visual contents to search images from large scale image databases according to users’ interests, has been an active and fast advancing research area over the last few years, fuelled at least in part by the rapid growth of imaging on the World-Wide Web. Users are exploiting the opportunity to access remotely-stored images in all kinds of new and exciting ways [Gudivada1995a]. However, this has exacerbated the problem of locating a desired image in a large and varied collection. This has led to the rise of a new research and development area - content-based image retrieval (CBIR), the retrieval of images on the basis of features automatically extracted from the images themselves.

The Internet is one of the best places to find different types of data such as images, text documents etc. But due to nuances of natural language, it is very difficult to get relevant information. For example, if the user is looking for sunset images, he may get different types of images. Only a few of them satisfy the users’ interest because the existing software retrieve images on the basis of string match and many of these are completely irrelevant to the user [Enser1995]. To handle such types of ambiguities, we develop feedback based techniques for customized content-based image retrieval.

1.2 Growth of Digital Imaging

The use of images in human communication is hardly new. Our cave-dwelling ancestors painted pictures on the walls of their caves, and the use of maps and building plans to convey information almost certainly dates back to pre-Roman times. But the
twentieth century has witnessed unparalleled growth in the number, availability and importance of images in all walks of life. Images now play a crucial role in fields as diverse as medicine, journalism, advertising, design, education and entertainment [Eakins 1999]. Technology, in the form of inventions such as photography and television, has played a major role in facilitating the capture and communication of image data. But the real engine of the imaging revolution has been the computer, bringing with it a range of techniques for digital image capture, processing, storage and transmission. The involvement of computers in imaging soon penetrated into areas traditionally dependent heavily on images for communication, such as engineering, architecture and medicine. Photograph libraries, art galleries and museums, too, began to see the advantages of making their collections available in electronic form. The creation of the World-Wide Web in the early 1990s, enabling users to access data in a variety of media from anywhere on the planet, has provided a further massive stimulus to the exploitation of digital images [Eakins1999].

1.3 Need for Image Data Management

The process of digitization does not in itself make image collections easier to manage. Some form of cataloging and indexing is still necessary to retrieve and view the desired pictures. The only benefit of digitization of images is that much of the required information can now potentially be derived automatically from the images themselves. However, significant research advances, involving collaboration between a number of disciplines, would be needed before image providers could take full advantage of the opportunities offered [Jain 1993]. There are a number of critical areas where research is
needed, including data representation, feature extractions and indexing, image query matching and user interfacing [Eakins1999].

One of the main problems is the difficulty of locating a desired image in a large and varied collection. While it is perfectly feasible to identify a desired image from a small collection simply by browsing, more effective techniques are needed with collections containing thousands of items. Journalists requesting photographs of a particular type of event, designers looking for materials with a particular color or texture, and engineers looking for drawings of a particular type of part, all need some form of access by image content [Jain 1993].

1.4 Characteristics of Image Queries

Access to a desired image from a repository might thus involve a search for images depicting specific types of objects or scenes, evoking a particular mood, or simply containing a specific texture or pattern. Potentially, images have many types of attribute which could be used for retrieval, including [Eakins 1999]:

a) Presence of a particular combination of color, texture or shape features (e.g. green stars).

b) Presence or arrangement of specific types of object (e.g. chairs around a table).

c) Depiction of a particular type of event (e.g. a football match).

d) Presence of named individuals, locations, or events (e.g. the Queen greeting a crowd).

c) Subjective emotions one might associate with the image (e.g. happiness).

f) Metadata such as who created the image, where and when.
Each listed query type represents a higher level of abstraction than its predecessor, and each is more difficult to answer without reference to some body of external knowledge [Eakins 1999].

1.5 **Current Content-Based Image Retrieval Techniques**

The currently used Content-Based Image Retrieval techniques retrieves stored images from a collection of given images by comparing features automatically extracted from the images themselves. The most common features used are mathematical measures of color, texture or shape. A typical system allows users to formulate queries by submitting an example of the type of image being sought, though some offer alternatives such as selection from a palette or sketch input. The system then identifies those stored images whose feature values match those of the query most closely, and displays thumbnails of these images on the screen [Long 2003]. A generic CBIR system is shown in Figure 1.1. Some of the more commonly used features for image retrieval are described below.

![Diagram for Content Based Image Retrieval System](image-url)

*Figure 1.1 Diagram for Content Based Image Retrieval System [Long 2003].*
1.5.1 Color Retrieval

Several methods for retrieving images on the basis of color are available in the literature, but most are variations on the same basic idea. Each image added to the collection is analyzed to compute a color histogram which shows the proportion of pixels of each color within the image. The color histogram for each image is then stored in the database. At search time, the user can either specify the desired proportion of each color (75% olive green and 25% red, for example), or submit an example image from which a color histogram is calculated. Either way, the matching process then retrieves those images whose color histograms match those of the query most closely [Eakins 1999].

1.5.2 Texture Retrieval

The ability to retrieve images on the basis of texture similarity may not seem very useful. But the ability to match on texture similarity can often be useful in distinguishing between areas of images with similar color (such as sky and sea, or leaves and grass). A variety of techniques has been used for measuring texture similarity; the best-established rely on comparing values of what are known as second-order statistics calculated from query and stored images. Essentially, these calculate the relative brightness of selected pairs of pixels from each image. From these it is possible to calculate measures of image texture such as the degree of contrast, coarseness, directionality and regularity [Tamura 1978], or periodicity, directionality and randomness [Liu 1996]. Texture queries can be formulated in a similar manner to color queries, by selecting examples of desired textures from a palette, or by supplying an example query image. The system then retrieves images with texture measures most similar in value to the query [Eakins 1999].
1.5.3 Shape Retrieval

The ability to retrieve by shape is perhaps the most obvious requirement at the primitive level. Unlike texture, shape is a fairly well-defined concept and there is considerable evidence that natural objects are primarily recognized by their shape [Biederman 1987]. A number of features characteristic of object shape which are independent of size or orientation of an object is computed for every object identified within each stored image. Queries are then answered by computing the same set of features for the query image, and retrieving those stored images whose features most closely match those of the query. Two main types of shape feature are commonly used – global features such as aspect ratio, circularity and moment invariants [Niblack 1993] and local features such as sets of consecutive boundary segments [Mehrotra 1995]. Queries to shape retrieval systems are formulated either by identifying an example image to act as the query, or as a user-drawn sketch [Hirata 1992].

Shape matching of three-dimensional objects is a more challenging task particularly where only a single 2-D view of the object in question is available. While no general solution to this problem is possible [Chen 1996], some useful inroads have been made into the problem of identifying at least some instances of a given object from different viewpoints. One approach has been to build up a set of plausible 3-D models from the available 2-D image, and match them with other models in the database [Chen 1996]. Another is to generate a series of alternative 2-D views of each database object, each of which is matched with the query image [Dickinson 1998]. Related research issues in this area include defining 3-D shape similarity measures [Shum 1996], and providing a means for users to formulate 3-D shape queries [Horikoshi 1990].
1.5.4 Retrieval by Other Types of Primitive Feature

One of the oldest-established means of accessing pictorial data is retrieval by its position within an image. Accessing data by spatial location is an essential aspect of geographical information systems, and efficient methods to achieve this have been around for many years [Roussopoulos 1988]. Similar techniques have been applied to image collections, allowing users to search for images containing objects in defined spatial relationships with each other [Chang 1988]. Improved algorithms for spatial retrieval are still being proposed [Gudivada 1995]. Spatial indexing is seldom useful on its own, though it has proved effective in combination with other cues such as color [Stricker 1996] and shape [Hou 1992].

Several other types of image features have been proposed as a basis for CBIR. Most of these rely on complex transformations of pixel intensities which have no obvious counterpart in any human description of an image. Most such techniques aim to extract features which reflect some aspect of image similarity which a human subject can perceive, even if he or she finds it difficult to describe. The most well researched technique of this kind uses the wavelet transform to model an image at several different resolutions. Promising retrieval results have been reported by matching wavelet features computed from query and stored images [Liang 1998]. Another method giving interesting results is retrieval by appearance. Two versions of this method have been developed, one for whole-image matching and one for matching selected parts of an image [Earkins 1999]. The part-image technique involves filtering the image with Gaussian derivatives at multiple scales [Ravela 1998a], and then computing differential invariants; the whole-image technique uses distributions of local curvature and phase [Ravela 1998b].
The advantage of all these techniques is that they can describe an image at varying levels of detail which can be useful enough in natural scenes where the objects of interest may appear in a variety of guises, and avoid the need to segment the image into regions of interest before shape descriptors can be computed. Despite recent advances in techniques for image segmentation [Campbell 1997], this remains a troublesome problem [Earkin 1999].

1.6 Objective of the Project

The area of content-based image retrieval is a hybrid research area that requires knowledge of both computer vision and of database systems. Large image databases are being collected, and images from these collections made available to users in advertising, marketing, entertainment, and other areas where images can be used to enhance the product. These images are generally organized loosely by category, such as animals, natural scenes, people, and so on. Human indexers who list the important objects in an image and other terms by which users may wish to access it do all image indexing. This method is not suitable for today's very large image databases [Dey 2003].

Content-based retrieval systems utilize measures that are based on low-level attributes of the image itself, including color histograms, color composition, and texture [Long 2003]. State-of-the-art research focuses on more powerful measures that can find regions of an image corresponding to known objects that users wish to retrieve. There has been some success in finding human faces of different selected sizes, human bodies, horses, zebras and other texture animals with known patterns, and such backgrounds as jungles, water, and sky [Dey 2003]. Our research work focuses on a unified methodology
for feature representation and object class recognition. The developed system is an image retrieval system that utilizes the layout and the structure of the perceptually correct color within an image to measure the similarity of images and then eventually classify these retrieved images into different categories using rough set based classification techniques.

1.7 Application Areas of Content Based Image Retrieval

A wide range of possible applications for CBIR technology has been identified. Some potentially fruitful areas are described in the following subsections:

1.7.1 Crime Prevention

Law enforcement agencies typically maintain large archives of visual evidence, including past suspects’ facial photographs, fingerprints, tire treads and shoeprints. Whenever a serious crime is committed, they can compare evidence from the scene of the crime for its similarity to records in their archives.

1.7.2 The Military

Military applications of imaging technology are probably the best-developed, though least publicized. Recognition of enemy aircraft from radar screens, identification of targets from satellite photographs, and provision of guidance systems for cruise missiles are known examples. Many of the surveillance techniques used in crime prevention could also be relevant to the military field [Eakins 1999].

1.7.3 Intellectual Property

Trademark image registration, where a new candidate mark is compared with existing marks to ensure that there is no risk of confusion, has long been recognized as a
prime application area for CBIR. Copyright protection is also a potentially important application area.

1.7.4 Architectural and Engineering Design

Architectural and engineering design shares a number of common features. The ability to search design archives for previous examples which are in some way similar, or meet specified suitability criteria, can be valuable [Eakins 1999].

1.7.5 Fashion and Interior Design

Similarities can also be observed in the design process in other fields, including fashion and interior design. The ability to search a collection of fabrics to find a particular combination of color or texture is increasingly being recognized as a useful aid to the design process [Eakins 1999].

1.7.6 Journalism and Advertising

Both newspapers and stock shot agencies maintain archives of still photographs to illustrate articles or advertising copy. These archives can often be extremely large (running into millions of images), and dauntingly expensive to maintain if detailed keyword indexing is provided. Broadcasting corporations are faced with an even bigger problem, having to deal with millions of hours of archive video footage, which are almost impossible to annotate without some degree of automatic assistance. This application area is probably one of the prime users of CBIR technology at present.

1.7.7 Medical Diagnosis

The increasing reliance of modern medicine on diagnostic techniques such as radiology, histopathology, and computerized tomography has resulted in an explosion in
the number and importance of medical images now stored by most hospitals. While the prime requirement for medical imaging systems is to be able to display images relating to a named patient, there is increasing interest in the use of CBIR techniques to aid diagnosis by identifying similar past cases [Eakins 1999].

1.7.8 Geographical Information Systems (GIS) and Remote Sensing

Although not strictly a case of image retrieval, managers responsible for planning marketing and distribution in large corporations need to be able to search by spatial attribute (e.g. to find the 10 retail outlets closest to a given warehouse). And the military are not the only group interested in analyzing satellite images. Agriculturalists and physical geographers use such images extensively, both in research and for more practical purposes, such as identifying areas where crops are diseased or lacking in nutrients – or alerting governments to farmers growing crops on land they have been paid to leave lying fallow [Eakins 1999].

1.7.9 Cultural Heritage

Museums and art galleries deal in inherently visual objects. The ability to identify objects sharing some aspect of visual similarity can be useful both to researchers trying to trace historical influences, and to art lovers looking for further examples of paintings or sculptures appealing to their taste [Eakins 1999].

1.7.10 Education and Training

It is often difficult to identify good teaching material to illustrate key points in a lecture or self-study module. The availability of searchable collections of video clips providing examples of (say) avalanches for a lecture on mountain safety, or traffic
congestion for a course on urban planning, could reduce preparation time and lead to improved teaching quality. In some cases (complex diagnostic and repair procedures) such videos might even replace a human tutor [Eakins 1999].

1.7.11 Home Entertainment

Much home entertainment is image or video-based, including holiday snapshots, home videos and scenes from favorite TV programmes or films. This is one of the few areas where a mass market for CBIR technology could develop [Eakins 1999].

1.7.12 Web Searching

The well-publicized difficulty of locating images on the Web [Jain 1995] indicates that there is a clear need for image search tools of similar power. Several experimental systems for content-based image searching on the Web have been demonstrated over the last two to three years.
2. NARRATIVE

2.1 Current Research Trends

Many image retrieval systems can be conceptually described by the framework depicted in Figure 2.1. The user interface typically consists of a query formulation part and a result presentation part. Specification of which images to retrieve from the database can be done in many ways. One way is to browse through the database one by one. Another way is to specify the image in terms of keywords, or in terms of image features that are extracted from the image, such as a color histogram. Yet another way is to provide an image or sketch from which features of the same type must be extracted as for the database images, in order to match these features.

A nice taxonomy of interaction models is given in [Vendrig 1997]. Relevance feedback is about providing positive or negative feedback about the retrieval result, so that the system can refine the search. There are a number of different classes of features available that are used to specify queries: color, texture, shape, spatial layout, and faces. Color features are often easily obtained directly from the pixel intensities, e.g. color histogram over the whole image, over a fixed sub image, or over a segmented region is often used.

Although a precise definition of texture is untraceable, the notion of texture generally refers to the presence of a spatial pattern that has some properties of homogeneity. In particular, the homogeneity cannot result from the presence of only a single color in the regions, but requires interaction of various colors.

There is no representation can be derived. Shape descriptors are diverse, e.g. turning angle functions, deformable templates, algebraic moments, and Fourier
coefficients [Remco 2001]. Indexing is often used as identifying features within an image; with indexing data structures speed up the retrieval of features within image collections. Spatial layout is about the absolute or relative position of color, texture, or shape information. Higher level features are increasingly more specific, and thus less widely used. However, faces are frequently present in pictures and relatively often used as a feature [Remco 2000]. Universal definition of what shape is either. Impressions of shape can be conveyed by color or intensity patterns, or texture, from which a geometrical.

Figure 2.1 Content Based Image Retrieval Framework [Remco 2000].
2.2 Research and Developed System

Here, we have developed an approach for learning user preference in personalized content-based image retrieval systems. This approach makes use of the relevance feedback from the user to categorize the target images from a given database. Relevance feedback is a technique that takes advantage of human-computer interaction to refine high level queries represented by low level features [Rui 1998]. This concept is used very much in traditional document retrieval systems (text based) for automatically adjusting an existing query using information fed back from the user [Rui 1998].

In the developed application of image retrieval, a user selects relevant images from previous retrieved results and provides a preference weight for each relevant image. The weights for the low-level feature, i.e., color and texture, etc., are then dynamically updated based on the user’s feedback. The user is no longer required to specify a precise weight for each low-level feature at the query formulation stage.

Based on the user’s feedback, the high level concepts implied by the query weights are automatically refined. Here we have implemented the concept of rough set theory which is described in detail in appendix A. During the process of relevant feedback, the similarity between the query (relevant images) and those in the database can then be calculated. Furthermore the user preferences can be stored in a personal profile which can be used in future for further retrieval. The effectiveness of the developed system has been supported with some experimental results. A generic model for the proposed system is illustrated in Figure 2.2.
A set of low-level visual features \( F = [f_i] \) (e.g. color, texture, shape) are used to characterize the input image \( I \). In the developed system, we use image histograms for the characterization of the input images. The image histogram counts for each possible image intensity (for instance 0-255). How many pixels are there in the image with that particular intensity, each feature, such as color, texture or shape, may be represented in several different ways. For each representation of a given feature \( r_{ij} \), a numeric description \( d_{ij} \) of the feature is extracted and stored. This description may be a histogram, a matrix of values or a feature vector of discriminating parameters. A similarity measure \( s_{ij} \) is associated with each feature description, and a weight \( w_{ij} \) indicating the significance of the particular description is assigned. In the developed system we use RGB values of each pixel in a given image to calculate the intensity histogram of each color. In the RGB color model red, green, and blue are used as the three primary colors. Quantifying the similarity between a query image and each image in a database is then calculated using
the concept of rough set theory (refer appendix A). Most similarity measures have a
range [0 1], where 1 indicates a perfect match and 0 indicates complete dissimilarity. The
total similarity between the query image q and an image in the database x is calculated by
adding the weighted similarities for each descriptor [Buckingham 2002].
3. System Design and Implementation

3.1 Fundamental of Images

3.1.1 Definition

A digital image $a[m, n]$ described in a two dimensional discrete space is derived from an analog image $a(x, y)$ in a two dimensional continuous space through a sampling process that is frequently referred to as digitization. The effect of digitization is shown in Figure 3.1. The two dimensional continuous image $a(x, y)$ is divided into $N$ rows and $M$ columns. The intersection of a row and a column is termed a pixel. The value assigned to the integer coordinates $[m, n]$ with $\{m = 0,1,2,...,M-1\}$ and $\{n = 0,1,2,...,N-1\}$ is $a[m, n]$. In fact, in most cases $a(x, y)$ which we might consider to be the physical signal that impinges on the face of a two dimensional sensor is actually a function of many variables including depth ($z$), color ($\lambda$), and time ($t$) [Now 2005].

![Figure 3.1 Digitization of a Continuous Image [Now 2005].](image)

The image shown in Figure 3.1 has been divided into $N = 16$ rows and $M = 16$ columns. The value assigned to every pixel is the average brightness in the pixel rounded to the nearest integer value. The process of representing the amplitude of the 2D signal at
a given coordinate as an integer value with L different gray levels is usually referred to as amplitude quantization or simply quantization [Gonzalez 2001].

3.1.2 RGB Color Model

In the RGB color model, red, green, and blue are used as the three primary colors. We don't actually specify what wavelengths these primary colors correspond to, so this will be different for different types of output media, e.g., different monitors, film, videotape, slides, etc. We can represent the RGB model by using a unit cube. Each point in the cube or the vector where the other point is the origin represents a specific color. This model is the best for setting the electron guns for a CRT. Note that for the "complementary" colors the sum of the values equals white light (1, 1, 1). For example: red (1, 0, 0) + cyan (0, 1, 1) = white (1, 1, 1) or green (0, 1, 0) + magenta (1, 0, 1) = white (1, 1, 1). Similarly blue (0, 0, 1) + yellow (1, 1, 0) = white (1, 1, 1) [Dey 2003].
a given coordinate as an integer value with \( L \) different gray levels is usually referred to as amplitude quantization or simply quantization [Gonzalez 2001].

### 3.1.2 RGB Color Model

In the RGB color model, red, green, and blue are used as the three primary colors. We don't actually specify what wavelengths these primary colors correspond to, so this will be different for different types of output media, e.g., different monitors, film, videotape, slides, etc. We can represent the RGB model by using a unit cube. Each point in the cube or the vector where the other point is the origin represents a specific color. This model is the best for setting the electron guns for a CRT. Note that for the "complementary" colors the sum of the values equals white light \((1, 1, 1)\). For example: red \((1, 0, 0)\) + cyan \((0, 1, 1)\) = white \((1, 1, 1)\) or green \((0, 1, 0)\) + magenta \((1, 0, 1)\) = white \((1, 1, 1)\). Similarly blue \((0, 0, 1)\) + yellow \((1, 1, 0)\) = white \((1, 1, 1)\) [Dey 2003].

![RGB Color Model](image)

**Figure 3.2 RGB Color Model [Dey 2003]**
3.1.3 Histograms

The image histogram counts for each possible image intensity (for instance 0–255) how many pixels are there in the image with a particular intensity. Peaks in the histogram represent frequent pixel intensities, and can often be related to nearly homogeneous regions. Figure 3.3 shows a histogram sample for the given picture. The X axis represents pixel values and the Y axis represents their frequencies.

![Histogram](image)

**Figure 3.3 Roses and its Histogram.**

3.2 CBIR Using a Rough Set with Heuristics

Rough sets theory was introduced by Pawlak in the early 1980s as a mathematical tool to deal with uncertainty. It is an extension of set theory for the study of the intelligent systems characterized by insufficient and incomplete information. Reduct is an important concept in rough sets theory. Reduct may be defined as those minimal attribute sets of information, which keep the same classification capability as the original attribute set.
contains [Pawlak 1982]. The aim of feature subset selection is to find a minimum set of relevant attributes that describe the dataset as well as the original all attributes do [Pawlak 1982]. So finding a reduct is similar to feature selection. Rough sets theory provides an approach to find out all reduct (all possible feature subsets) [Pawlak 1982]. The concept of rough set theory is explained more in appendix A.

The term heuristic is used for algorithms which find solutions among all possible ones, but they do not guarantee that the best will be found, therefore they may be considered as approximately and not accurate algorithms. These algorithms, usually find a solution close to the best one and they find it fast and easily. Sometimes these algorithms can be accurate, that is they actually find the best solution, but the algorithm is still called heuristic until this best solution is proven to be the best [Nikolaos, 2003]. The method used from a heuristic algorithm is one of the known methods, such as greediness. Concept of heuristic algorithm with some examples is more explained in appendix A.

### 3.3 Developed System Design

We have already stated that image retrieval has become an important technology. It is widely used in search engines, object recognition etc. Also we have taken an overview of information retrieval using rough set theory (appendix A). We have implemented the same in our image retrieval system.

#### 3.3.1 Histogram Generation & Pixel Grabber

As shown with the data flow diagram in Figure 3.4. At first we create the histograms and see the nature of the sample images (Images which act as the training set.
These images have already been categorized by the user in different groups. For each image we calculate the percentage of pixels that lie in a particular range. The percentage of pixels is calculated using the following formula:

\[
\left( \frac{\text{No. of pixels in the given range}}{\text{Total no. of pixels}} \right) \times 100
\]

Every image is a combination of red, green and blue. These colors give meaningful information about an image. In our developed system, these colors act as attributes for the images. Each color range (0-255) is divided into eight different intervals and each interval has a span of 32 pixels. The intervals are as follows for each color (0-31, 32-63… 224-255). For all these sample images, the user has already defined the category to which each image belongs. In our system we represent different categories defined by the user with numbers. As a result each category is assigned a unique number.

Figure 3.4 Level 1 Data Flow Diagram
The Table 3.1 shows the decision table obtained after calculating the percentage of pixels that lie in each range for the RGB values. In this table, columns represent the attributes (percentage of RGB values) of the pixels range-wise. The last column in the table represents the category for the image which was defined by the user. Each row represents a unique image.

<table>
<thead>
<tr>
<th>Image</th>
<th>R(0-31)</th>
<th>...</th>
<th>R(224-225)</th>
<th>...</th>
<th>G(0-31)</th>
<th>...</th>
<th>G(224-255)</th>
<th>B(0-31)</th>
<th>...</th>
<th>B(224-255)</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Img1</td>
<td>12</td>
<td>...</td>
<td>4</td>
<td>33</td>
<td>...</td>
<td>12</td>
<td>4</td>
<td>0</td>
<td>...</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Img2</td>
<td>34</td>
<td>...</td>
<td>25</td>
<td>23</td>
<td>...</td>
<td>15</td>
<td>23</td>
<td>31</td>
<td>...</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>ImgN</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 3.1 Decision Table for Training Set

3.3.2 Heuristics Techniques and Rough Set Comparison

Our Content based retrieval approach depends on attribute values for each color. In rough set based methodology, we make a decision system where each column is represented by pixel intervals and each row is represented by an image. Using the discernibility matrix (appendix A), we find a minimal set of cuts for different colors which can discern maximal numbers of images. To find the minimal set of cuts, we apply the heuristic algorithm on the discernibility matrix (appendix A). It gives the cut values for different colors and weights for different images. This approach reduces the size of the decision system by considering only those intervals which are generated by the heuristic technique. Initially, we have 24 different cuts but after applying heuristic approach we have to consider only seven different cut values. After building the decision
system based on above approach we then input the set of uncategorized images (set of new images) for which we need to make the decision. A table similar to Table 3.1 is then generated for new images without the category column. The above calculated cut values (pixel values) are then used to make the comparison, between categorized and uncategorized images. The corresponding value of each image with respect to these cut values are used in the calculation of rough membership. Finally we calculate the rough membership of new images using the membership formulae. The new image is assigned the category for which the image has the maximum rough membership. The categorized images are then stored in the database as shown in the Figure 3.5.

![Figure 3.5 Level 1 Data Flow Diagram](image-url)
3.3.3 Membership function for Image grading

In this section we present the membership function to compute the ranks of new images. The degree of membership of an element $x$ to a category $C$, is given by the overlap between the set of elements in category $C$ in the given information table, and the equivalence class $[x]_A$ to which $x$ belongs, where $A$ is the set of description attributes.

The final categorization of an element is the category for which the magnitude of its membership function is maximum.

$$
\mu^A_C : U \rightarrow [0, 1], \text{ and } \mu^A_C (x) = \frac{|[x]_A \cap C|}{|[x]_A|} \quad (1)
$$

This function takes into account the relative degree of membership of a document into different categories with respect to each reduct and then provides a final categorization of the new document as a function of all these memberships.

For a new document, for each RGB interval present in $W$, the weight of the RGB interval in the document is determined and normalized. Then the interval to which the weight belongs to, on the basis of reducts obtained earlier is determined. The new image membership to each category $C$, denoted by $\mu_C (x)$, is computed as follows. Let $a_{s^*}(x)$ denote the weight of $s^*$ or one of its synonyms in document $x$. 
\[ [s^*] = \{(x, x') \in U^2 | \forall s^* \in W, a_\gamma(x) = a_\gamma(x')\} \]

\[ \mu_{c}^{\star}(x) = \frac{|[s^*] \cap C|}{|[s^*]|}, \text{where } C = (1, 2, 3), \quad (2) \]

\[ \forall \ s^* \in W, \text{ provided } |[s^*]| \neq \varnothing \cdot \mu_{c}^{\star}(x) \in [0, 1] \]

\[ \mu_{c}(x) = \left[ \sum_{j} \mu_{c}^{j}(x) \right], \text{where } (1 \leq j \leq p), \text{ and } p = \text{no.of most discerning words} \quad (3) \]

\[ \mu_{c}^{\star}(x) = \left( \frac{\mu_{c}(x)}{\sum \mu_{c}(x)} \right), \text{Decision} = \left\{ d : \max_{d} \left( \mu_{d}^{\star}(x) \right) \right\} \quad (4) \]

Equation 2 computes the membership of an image for each category \( X \) by taking into account the relevance of each discerning RGB values \( s^* \) for that category. Equation 3 computes the total membership value for a single category for the new document for the complete set of discerning RGB values. Equation 4 computes the final membership value of an image \( x \) to a category \( C \), by normalizing it against all membership values. The category with the maximum weight is assigned to the image.
4. Results and Discussions

41. Results Obtained

Here we have taken following three categories of training set of images from the website www.google.com. The decision values or the category value for each set is given in bracket. These images are used by our software as the training images.

Table 4.1 Category: Neon Images (Decision value 1)

<table>
<thead>
<tr>
<th>Neon1</th>
<th>Neon2</th>
<th>Neon3</th>
<th>Neon4</th>
<th>Neon5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neon6</td>
<td>Neon7</td>
<td>Neon8</td>
<td>Neon9</td>
<td>Neon10</td>
</tr>
</tbody>
</table>

Table 4.2 Category: Rose Images (Decision value 2)
Given below is the set of uncategorized images. Our system is going to categorize these images into different categories.
Shown below are the membership value calculated by the system, for new images.

<table>
<thead>
<tr>
<th>Images</th>
<th>Membership Value</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neon11</td>
<td>0.6111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.12777779</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.2611111</td>
<td>3</td>
</tr>
<tr>
<td>Neon12</td>
<td>0.1111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.5777778</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.31111112</td>
<td>3</td>
</tr>
<tr>
<td>Neon13</td>
<td>0.44444445</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.41111112</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.14444445</td>
<td>3</td>
</tr>
<tr>
<td>Neon14</td>
<td>0.2777778</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.29444444</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.42777777</td>
<td>3</td>
</tr>
<tr>
<td>Neon15</td>
<td>0.2777778</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.29444444</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.42777777</td>
<td>3</td>
</tr>
<tr>
<td>Neon16</td>
<td>0.6111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.12777779</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.2611111</td>
<td>3</td>
</tr>
<tr>
<td>Images</td>
<td>Membership Value</td>
<td>Category</td>
</tr>
<tr>
<td>--------</td>
<td>------------------</td>
<td>----------</td>
</tr>
<tr>
<td></td>
<td>0.27777778</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.29444444</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.42777777</td>
<td>3</td>
</tr>
<tr>
<td>Rose11</td>
<td>0.11111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.57777778</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.31111112</td>
<td>3</td>
</tr>
<tr>
<td>Rose12</td>
<td>0.11111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.57777778</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.31111112</td>
<td>3</td>
</tr>
<tr>
<td>Rose13</td>
<td>0.11111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.57777778</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.31111112</td>
<td>3</td>
</tr>
<tr>
<td>Rose14</td>
<td>0.11111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.57777778</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.31111112</td>
<td>3</td>
</tr>
<tr>
<td>Rose15</td>
<td>0.27777778</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.29444444</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.42777777</td>
<td>3</td>
</tr>
<tr>
<td>Rose16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Images</td>
<td>Membership Value</td>
<td>Category</td>
</tr>
<tr>
<td>----------</td>
<td>------------------</td>
<td>----------</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.275</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.475</td>
<td>3</td>
</tr>
<tr>
<td>Rose17</td>
<td>0.11111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.5777778</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.31111112</td>
<td>3</td>
</tr>
<tr>
<td>Animal11</td>
<td>0.25</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.275</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.475</td>
<td>3</td>
</tr>
<tr>
<td>Animal12</td>
<td>0.11111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.5777778</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.31111112</td>
<td>3</td>
</tr>
<tr>
<td>Animal13</td>
<td>0.44444445</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.41111112</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.14444445</td>
<td>3</td>
</tr>
<tr>
<td>Animal14</td>
<td>0.2777778</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.29444444</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.42777777</td>
<td>3</td>
</tr>
<tr>
<td>Animal15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Images</td>
<td>Membership Value</td>
<td>Category</td>
</tr>
<tr>
<td>---------</td>
<td>------------------</td>
<td>----------</td>
</tr>
<tr>
<td></td>
<td>0.2777778</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.294444444</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.427777777</td>
<td>3</td>
</tr>
<tr>
<td>Animal16</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.6111111</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>0.12777779</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>0.2611111</td>
<td>3</td>
</tr>
<tr>
<td>Animal17</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 4.5 Membership Values for Uncategorized Images**

Above values shows in which category the image belongs. A higher value in a set shows the high resemblance in that category. The following table shows the rating given by the user and the rating given by the system.

<table>
<thead>
<tr>
<th>Images</th>
<th>Rating given by the user</th>
<th>Rating given by the system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neon11</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Neon12</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Neon13</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Neon14</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Neon15</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Neon16</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Rose11</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Rose12</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Rose13</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Rose14</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Rose15</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Rose16</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Rose17</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Animal11</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Animal12</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Animal13</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Animal14</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Animal15</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Animal16</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Animal17</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

**Table 4.6 Rating Given to New Images by the System**

The above table shows out of twenty images ten are retrieved correctly. The retrieval correctness of the system is 50%. This efficiency will increase with inclusion of more number of images and with the consideration of other attributes of the images, for example brightness, shape, texture etc.
5. TESTING AND EVALUATION

5.1 System Performance Metrics

The objective of this information retrieval system is to retrieve as many relevant items as possible meanwhile to reject as many irrelevant items as possible. A good effectiveness measurement should indicate the ability of meeting this objective [Sajjanhar 1997]. The retrieval effectiveness becomes one of the most important parameters to measure the performance of content-based image retrieval systems. The performance of this proposed content based image retrieval system application was measured by testing the system with different sets of image databases. Images from entirely different databases were collected and then tested with the developed system. Thus measuring how the accurate our proposed system is in retrieval of the required set of images.

5.2 Unit Testing

This type of testing is done to determine that individual program modules perform to specification. Each module is tested alone in an attempt to discover any errors in its code. It is ideal to develop the software applications as components and be able to dynamically test each component individually by considering the input/outputs [Hunt, 2003].

5.3 Regression Testing

Test everything tested before. This is the only way to validate whether the new amendments to the software have introduced any flaws to the functionality working previously.
Finally after the system was developed and tested with different sets of images it was calculated that the retrieval correctness of the system is 62%. This efficiency will increase with inclusion of more images and also with the consideration of other attributes of the images.
6. CONCLUSION

6.1 Conclusion

In the developed system, we have proposed rough set based techniques for content based image retrieval. This approach handles uncertainty, vagueness of image data very elegantly. In this project, image color pixel values are used as an image attribute and discernibility based method to find out minimal set of pixel value intervals for different colors. This reduced set acts as the input to rough membership based computation to classify different class of images. Using this method, we have tried to work on different types of images and find the efficiency of our proposed system. The proposed system performs well to classify different category of images.

6.2 Limitations

In our developed system a color-histogram based low-dimensional technique has been implemented. Results obtained are more favorable for database of images, for which the user provides a rich set of training images. Also the system can categorize the given images only into the categories for which the system is trained.

Color histogram-based techniques are useful for matching images with similar color appearances, but this approach does not work very well for retrieving images in finer granularities. For retrieving images in finer granularity we have to use a combination of image properties along with color histograms.

The developed system can easily be extended to domains containing more colors and other image features like shape, texture and locations so as to overcome these limitations.
BIBLIOGRAPHY AND REFERENCES


Princeton, NJ, 82-87.


APPENDIX A

The content of the following Appendix A has been derived from “Rough Sets: A Tutorial” written by Komorowski, J., Polkowski, L., & Andrzej, S. Online link to the tutorial is http://www.let.uu.nl/esslli/Courses/skowron/skowron.ps.

1. Overview of Rough Set

An information system can be defined as a pair $D = (U, A)$, where $U$ is a non-empty finite set of objects called the universe and $A$ is a non-empty finite set of attributes such that $a: U \rightarrow V_a$ for every $a \in A$. The set $V_a$ represents the value set of attribute $a$. In many applications there is an outcome of classification that is known. This a posteriori knowledge is expressed by one distinguished attribute called decision attribute; the process is known as supervised learning. An information system is called a decision system if it has an additional decision attribute. This is defined as:

$$D = (U, A \cup \{d\})$$ where $d$ is the decision attribute

Example 1: This example illustrates an information system with seven objects, two attributes Age and Lower Extremity Motor Score (LEMS) and one decision attribute Walk with two possible outcomes.
<table>
<thead>
<tr>
<th>X</th>
<th>Age</th>
<th>LEMS</th>
<th>Walk</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>16 - 30</td>
<td>50</td>
<td>Yes</td>
</tr>
<tr>
<td>X2</td>
<td>16 - 30</td>
<td>0</td>
<td>No</td>
</tr>
<tr>
<td>X3</td>
<td>31 - 45</td>
<td>1 - 25</td>
<td>No</td>
</tr>
<tr>
<td>X4</td>
<td>31 - 45</td>
<td>1 - 25</td>
<td>Yes</td>
</tr>
<tr>
<td>X5</td>
<td>46 - 60</td>
<td>26 - 49</td>
<td>No</td>
</tr>
<tr>
<td>X6</td>
<td>16 - 30</td>
<td>26 - 49</td>
<td>Yes</td>
</tr>
<tr>
<td>X7</td>
<td>46 - 60</td>
<td>26 - 49</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 1: Walk: An example decision system

The decision to be synthesized from this information system can be stated as follows- “What are the attribute values for the Age and LEMS for which an object of the universe can walk?” It is required to analyze the attribute values for positive instances i.e., where “Walk = Yes” and negative instances i.e., where “Walk = No”. We can see that cases X3 and X4 as well as X5 and X7 have exactly the same values of attributes. The cases are (pair wise) indiscernible using the available attributes. However, the outcomes are not always same for indiscernible objects.

In the context of text information retrieval, the decision attribute is the gradation given to a document by a user. The decision system is built by storing the decisions given by the user on a set of training documents. The attributes of a document are the words present in it and their values indicate the relative importance of a word in a document. The core of all rough-set based reasoning contains an equivalence relation called the indiscernibility relation. For any B ⊆ A, the equivalence relation IND_D(B) is defined as:

\[ R = IND_D(B) = \{(x,x') \in U^2 \mid \forall b \in B, V_b(x) = V_b(x')\} \]  \hspace{1cm} (1)
This relation is called a B-indiscernibility relation. We denote the equivalence classes of the B-indiscernible relations as \([x]_B\). The subscript D in the indiscernibility relation can be omitted. In the context of text-information retrieval the equivalence relation that is generally used is the synonymy relation which establishes equivalence of two synonymous words. Thus two texts can be said to be roughly similar if they contain synonymous words but not necessarily the same words.

Example 2: The non-empty subsets of the conditional attributes of example one are \{Age\}, \{LEMS\} and \{Age, LEMS\}. The equivalence class for various subsets is given below.

\[
\text{IND} (\{\text{Age}\}) = \{\{x_1, x_2, x_6\}, \{x_3, x_4\}, \{x_5, x_7\}\} \\
\text{IND} (\{\text{LEMS}\}) = \{\{x_1\}, \{x_2\}, \{x_3, x_4\}, \{x_5, x_6, x_7\}\} \\
\text{IND} (\{\text{Age, LEMS}\}) = \{\{x_1\}, \{x_2\}, \{x_3, x_4\}, \{x_5, x_7\}, \{x_6\}\}
\]

The equivalence classes obtained from the indiscernibility relation are used to define set approximations.

2. Set Approximation

We find that in the Information System example, it is impossible to induce a precise description of \(x_3\) and \(x_4\). This is because though these objects are indiscernible with respect to \(A\), still the outcomes are different. It is here that the notion of Rough Set emerges. Rough Set Theory is applied for those elements in the universe that belong to a boundary between the certain cases. These notions are formally expressed as follows. Let \(D = (U, A)\) be an information system and let \(B \subseteq A\) and \(X \subseteq U\). We can approximate \(X\) using only the information contained in \(B\) by constructing the lower and upper approximations with respect to \(B\). Let \(U\) be represented by the collection of disjoint
equivalence classes with respect to the relation \( R \), i.e., \( U = \{ C_1, C_2, \ldots, C_n \} = [x]_R \). The pair \((U, R)\) is called an **approximation space**. The lower approximation of \( X \), denoted by \( \underline{\operatorname{apr}}_R (X) \) is defined by the set

\[
\underline{\operatorname{apr}}_R (X) = \{ x \in C_i \mid C_i \subseteq X \} \tag{2}
\]

and the upper approximation of \( X \), denoted by \( \overline{\operatorname{apr}}_R (X) \) is defined by the set

\[
\overline{\operatorname{apr}}_R (X) = \{ x \in C_i \mid C_i \cap X \neq \emptyset \} \tag{3}
\]

The objects in \( \underline{\operatorname{apr}}_R (X) \) can be **definitely** classified as members of \( X \) on the basis of the information in \( B \), while the objects in \( \overline{\operatorname{apr}}_R (X) \) can only be classified as **possible** members of \( X \). The boundary region of \( X \) is represented as

\[
\text{BN}_B (X) = \overline{\operatorname{apr}}_R (X) - \underline{\operatorname{apr}}_R (X) \tag{4}
\]

The boundary region contains those objects which cannot be definitely classified into \( X \) on the basis of the knowledge in \( B \). An **set is said to be rough (crisp) if this boundary region is non-empty**. The outside region of \( X \) is represented as:

\[
U - \overline{\operatorname{apr}}_R (X)
\]

The outside region of \( X \) consist of those objects which can be with certainty classified as do not belonging to \( X \) on the basis of the knowledge in \( B \).

Example 3: Let \( W = \{ x \mid \text{Walk} (x) = \text{Yes} \} \), as given by Table 1. We can obtain the approximation regions using the attributes Age and LEMS. It follows that the outcome Walk is rough since the boundary region is not empty. These lower approximation, upper approximation, boundary region, and outside region are shown below:
$A$-lower approximation of $W$:
\[ \underline{AW} = \{x_1, x_6\}, \]

$A$-upper approximation of $W$:
\[ \overline{AW} = \{x_1, x_3, x_4, x_6\}, \]

$A$-boundary region of $W$:
\[ BN_AW = \{x_3, x_4\}, \]

$A$-outside region of $W$:
\[ U - \overline{AW} = \{x_2, x_5, x_7\} \]

Figure 1: Approximating the set of walking patients.
3. Rough Membership

In classical set theory, either an element belongs to a set or it does not. The corresponding membership function is the characteristic function for the set, i.e., the function takes values 1 and 0, respectively. In the case of rough sets, the rough membership function quantifies the degree of relative overlap between the set $X$ and the equivalence $[x]_B$ class to which $x$ belongs. It is defined as follows:

$$
\mu^B_x : U \rightarrow [0,1] \text{ and }
$$

$$
\mu^B_x (x) = \frac{|[x]_B \cap X|}{|[x]_B|}
$$

4. Feature Extraction Methods

Discretization is a popular way to extract features doing partition of value sets of conditional attributes into intervals but selection of appropriate intervals and partitioning of attribute value sets is a complex problem and its complexity grows as the number of attributes grows. So partition is done in such a way that if name of interval containing an arbitrary object is substituted for any object instead of its original value a consistent decision system is also obtained.

In the discretization of a decision table $D$, where $V_a = [u_a, v_a]$ is an interval of real numbers, we search for a partition $P_a$ of $V_a$ for any $a \in A$. Any partition of $V_a$ is defined by a sequence of the so-called cuts $v_1 < v_2 < v_3 \ldots \ldots \ldots < v_r$ from $V_a$. Hence, any family of partitions $\{P_a\}_{a \in A}$ can be identified with a set of cuts. We consider an example to explain discretization process.
Example 4: Let us consider a (consistent) decision system with two conditional attributes $a$ and $b$ and seven objects $u_1, u_2, \ldots, u_7$. The values of decision $d$ are represented in Table 2. The sets of possible values of $a$ and $b$ are defined by $V_a = [0, 2)$ and $V_b = [0,4)$. The sets of values of $a$ and $b$ on objects from $U$ are given by $a(U) = \{0.8, 1.1, 3, 1.4, 1.6\}$ and $b(U) = \{0.5, 1, 2, 3\}$

<table>
<thead>
<tr>
<th>C</th>
<th>A</th>
<th>a</th>
<th>b</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>$U_1$</td>
<td>0.8</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>$U_2$</td>
<td>1</td>
<td>0.5</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>$U_3$</td>
<td>1.3</td>
<td>3</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>$U_4$</td>
<td>1.4</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>$U_5$</td>
<td>1.4</td>
<td>2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>$U_6$</td>
<td>1.6</td>
<td>3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>$U_7$</td>
<td>1.3</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Original Decision System

We will describe a discretization process that returns a partition of the value sets of conditional attributes into intervals. In the given example, the following intervals for conditional attributes are as follows: $[0.8, 1)$, $[1, 1.3)$, $[1.3, 1.4)$, $[1.4, 1.6)$ for $a$, and $[0.5, 1)$, $[1, 2)$, $[2, 3)$ for $b$. Let us introduce the idea of cuts. Cuts are pairs $(a, c)$ where $c \in V_a$ and $c$ is the middle points of the intervals defined above. Cuts for conditional attributes are defined as following:
Where \( p_1^a \sim (a, 0.9), p_2^a \sim (a, 1.15), p_3^a \sim (a, 1.35), \) and \( p_4^a \sim (a, 1.5) \). An analogous construction is done for \( b \). Cuts corresponding to \( b \) are \( p_1^b \sim (b, 0.75), p_2^b \sim (b, 1.5), \) and \( p_3^b \sim (b, 2.5) \).

Any cut defines a new conditional attribute with binary values. For example, the attribute corresponding to the cut \((a, 0.9)\) is equal to 0 if \( a(x) < 0.9 \), otherwise is equal to 1. Objects positioned on different sides of the straight line \( a=0.9 \) are discerned by this cut. The important question is: How to construct a set of cuts with minimal number of elements discerning all pairs of objects? To find it, we make Discernibility Matrix and apply MD heuristic method. This heuristic searches for cut which can discern maximal number of object pairs.

In the Discernibility matrix each row is represented by objects of different decisions and each column are represented by cut corresponding to each attribute and the value of the new attribute corresponding to a cut \((a, c)\) on the pair \((u_i, u_j)\) is equal to 1 if this cut is discerning objects \((u_i, u_j)\) (i.e. \( \min (a (u_i), a (u_j)) < c < \max (a (u_i), a (u_j)) \)) and 0 otherwise. Discernibility matrix of table 2 has shown as follows:
Table 3: Discernibility Matrix

We apply MD-Heuristic method on Discernibility matrix $D^*$. This searches for a cut with maximal number of objects pairs discerned by this cut. The steps for the implementation of heuristic method are defined below.

5. MD Heuristic Algorithm:

Step 1: Choose a column from Discernibility Matrix $D^*$ with maximal number of occurrences of 1’s;

Step 2: Delete from Discernibility Matrix the column chosen in step 1 and all rows which has value 1 for this column;

Step 3: If Discernibility Matrix is non-empty then go to step 1 else stop.

According to algorithm, first we choose $p_2^b$, next $p_2^a$ and finally $p_4^a$. Hence the resulting set of cuts $P = \{(a, 1.15), (a, 1.5), (b, 1.5)\}$ which can discern maximum numbers of objects. Using $P$, we define new conditional attributes $a_P$ for any $a$ and $b_P$ for any $b$. One should consider a partition of the value set of $a$ by cuts from $P$ and put the
unique names for the elements of these partition. We assign interval 0 for all values of $a$
less than 1.15, interval 1 for all values of $a$ in the interval [1.15, 1.5) and interval 2 for all
values of $a$ in the interval [1.5, 4). Similarly, we do for conditional attribute $b$. The
original decision system $D$ is transformed into $P$-discretization of $D$ which is given as
follows:

<table>
<thead>
<tr>
<th>C</th>
<th>A</th>
<th>a</th>
<th>b</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$U_1$</td>
<td>0.8</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$U_2$</td>
<td>1</td>
<td>0.5</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$U_3$</td>
<td>1.3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$U_4$</td>
<td>1.4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$U_5$</td>
<td>1.4</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$U_6$</td>
<td>1.6</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$U_7$</td>
<td>1.3</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>C</th>
<th>A</th>
<th>$a_p$</th>
<th>$b_p$</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$U_1$</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$U_2$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$U_3$</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$U_4$</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$U_5$</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$U_6$</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$U_7$</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4: Original Decision System and $P$-discretized System
APPENDIX B

Java is a pure object-oriented language. It supports image operations strongly. Java has many classes, interfaces and methods for operations on images. Some fundamental operations are displaying images, loading images, selecting and calculating pixel values etc. Java has a number of characteristic features that supports internet programming in which image operations are widely used. So we have used Java 2 Platform SE v1.4.2 in the development of our software.

Java contains AWT’s Image class and java.awt.image packages. Together they support imaging. Images are object of Image class which is part of java.awt package. Images are manipulated using the classes found in java.awt.image package. There are a large number of imaging classes and interfaces defined by java.awt.image.

1. Package java.awt.image Description

This package provides classes for creating and modifying images. Images are processed using a streaming framework that involves an image producer, optional image filters, and an image consumer. This framework makes it possible to progressively render an image while it is being fetched and generated. Moreover, the framework allows an application to discard the storage used by an image and to regenerate it at any time. This package provides a number of image producers, consumers, and filters that we can configure for our image processing needs. Define below are the classes and Interfaces used in the development of our content based image retrieval system.
## Class Summary

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ColorModel</td>
<td>The ColorModel abstract class encapsulates the methods for translating a pixel value to color components (for example, red, green, and blue) and an alpha component.</td>
</tr>
<tr>
<td>DirectColorModel</td>
<td>The DirectColorModel class is a ColorModel class that works with pixel values that represent RGB color and alpha information as separate samples and that pack all samples for a single pixel into a single int, short, or byte quantity.</td>
</tr>
<tr>
<td>FilteredImageSource</td>
<td>This class is an implementation of the ImageProducer interface which takes an existing image and a filter object and uses them to produce image data for a new filtered version of the original image.</td>
</tr>
<tr>
<td>ImageFilter</td>
<td>This class implements a filter for the set of interface methods that are used to deliver data from an ImageProducer to an ImageConsumer.</td>
</tr>
<tr>
<td>MemoryImageSource</td>
<td>This class is an implementation of the ImageProducer interface which uses an array to produce pixel values for an Image.</td>
</tr>
<tr>
<td>PixelGrabber</td>
<td>The PixelGrabber class implements an ImageConsumer which can be attached to an Image or ImageProducer object to retrieve a subset of the pixels in that image.</td>
</tr>
<tr>
<td>RGBImageFilter</td>
<td>This class provides an easy way to create an ImageFilter which modifies the pixels of an image in the default RGB ColorModel.</td>
</tr>
</tbody>
</table>

Table 1: Class Summary

## Interface Summary

<table>
<thead>
<tr>
<th>Interface</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageConsumer</td>
<td>The interface for objects expressing interest in image data through the ImageProducer interfaces.</td>
</tr>
<tr>
<td>ImageObserver</td>
<td>An asynchronous update interface for receiving notifications about Image information as the Image is constructed.</td>
</tr>
<tr>
<td>ImageProducer</td>
<td>The interface for objects which can produce the image data for Images.</td>
</tr>
</tbody>
</table>

Table 2: Interface Summary